

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
13 December 2001 (13.12.2001)

PCT

(10) International Publication Number
WO 01/95566 A2

(51) International Patent Classification⁷: **H04L 12/00**

(21) International Application Number: PCT/US01/18159

(22) International Filing Date: 5 June 2001 (05.06.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/209,413 5 June 2000 (05.06.2000) US

(71) Applicant (for all designated States except US): **QLOGIC SWITCH PRODUCTS, INC.** [US/US]; Suite 13, 6321 Bury Drive, Eden Prairie, MN 55346 (US).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **GEORGE, William, R.** [US/US]; 4744 Upton Avenue South, Minneapolis, MN 55410 (US). **DROPPS, Frank, R.** [US/US]; 17222-81st Avenue North, Maple Grove, MN 55311 (US).

(74) Agent: **VIKSINIS, Ann, S.**; Schwegman, Lundberg, Woessner & Kluth, P.O. Box 2938, Minneapolis, MN 55402 (US).

(81) Designated States (*national*): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States (*regional*): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

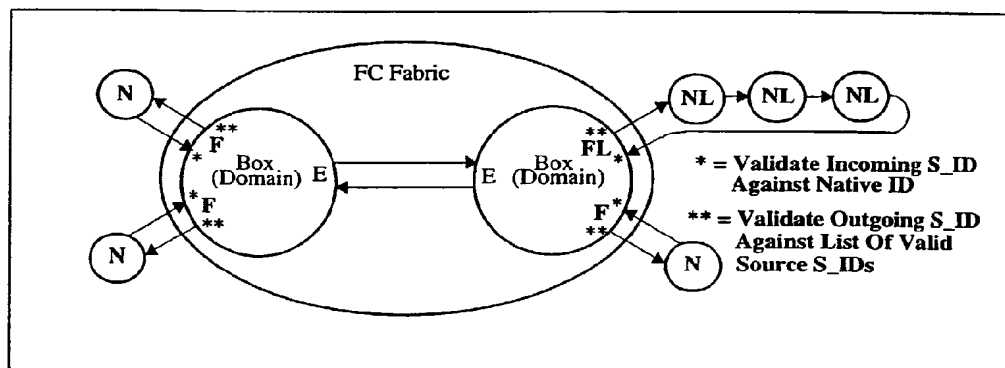
Published:

— without international search report and to be republished upon receipt of that report

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: **HARDWARE-ENFORCED LOOP-LEVEL HARD ZONING FOR FIBRE CHANNEL SWITCH FABRIC**

Fabric Cloud S_ID Validation Points



(57) **Abstract:** Hardware-enforced zoning is provided in Fibre Channel switches to protect against breaching of assigned zones in a switch network which can occur with software-based zoning techniques. The invention provides logic for performing a hardware-based validation of the Source ID S_ID of frames both at the point where the frame enters the Fibre Channel fabric, and at the point where the frame leaves the fabric. The S_ID is verified against an inclusion list or table of allowable S_IDs, which can be unique for each fabric port. The invention provides a way to increase the range of sources an inclusion table can express, by implementing wild cards, on an entry-by entry basis. If the S_ID is valid, it will enter the fabric and route normally. If invalid, the frame will not be routed but will be disposed of by the fabric according to the FC rules. This prevents incorrect S_IDs from breaching the table-driven zoning at the point where frames exit the fabric, to prevent unauthorized access to devices connected to the switch network.

WO 01/95566 A2

HARDWARE-ENFORCED LOOP-LEVEL HARD ZONING FOR FIBRE CHANNEL SWITCH FABRIC

Field of the Invention

This invention pertains generally to the field of high performance switching,
5 and in particular to improvements in Fibre Channel switching to provide the
establishment of zones for permitted access to connected devices, with hardware-
enforcement of the zoning.

Background of the Prior Art

The Fibre Channel standard has been established to provide for high
10 performance switching solutions for computing and data handling systems.
Examples of applications where the high speed and high bandwidth of Fibre
Channel switches may be used to advantage include interconnecting computers
and high-performance storage devices, interconnecting computers in multiple-
computer operating environments, and anywhere multiple high-speed data
15 interconnections must be established between designated nodes or groups of
nodes in data handling networks.

The Fibre Channel standard, ANSI X3.T11, broadly defines classes and
standards of function and performance, but does not dictate the implementation
technologies to be used in providing these functions. A particular design of a
20 switch to implement Fibre Channel functions is referred to as the 'fabric' of the
switch. As this invention is directed to improvements in Fibre Channel switches,
the description of the invention herein uses terminology and other defined terms
from the field of Fibre Channel (referred to by the notation "FC" below)
switches, and the FC standard may be consulted for definitions.

25 For data integrity and security reasons, it is necessary in some networks to
make certain that certain hosts or devices have controlled access. For example,
certain hosts may be allowed access to only certain storage devices, and vice
versa. This requirement means that certain channels or groups of channels to
which the affected hosts or devices are attached must be isolated from
30 communication to or from other channels or groups of channels. Zoning

techniques are used in prior art systems to define zones of addresses that will be considered valid for various sources or destinations connected to a switch.

Soft Zoning: The Problem

5 A problem with Fibre Channel zoning as it presently exists is that it is software-enforced zoning, often referred to as soft zoning. In soft zoning, devices connected to N_Ports and NL_Ports of the FC fabric login to the fabric and make queries of the Name Server to determine which of the remote devices this device can communicate with, along with their FC addresses (D_IDs). The Name Server defines and enforces the zones by listing in the login response the
10 set of devices (by D_ID) that are in the login requester's zone or zones. In this manner, devices honor zones by using only those D_IDs given out by the Name Server.

However, this works only if all devices follow the rules, and there are no hardware failures. Soft zoning can be breached in the following ways.

- 15 • Zones can be breached inadvertently by HBA software errors that generate incorrect D_IDs.
- Zones can be breached by hardware failures, where the D_ID is corrupted somewhere between the source device and the destination device.
- 20 • Zones can be breached deliberately by ill-mannered but non-malicious HBAs, such as those that walk through all D_IDs to discover where other HBAs are attached.
- Zones can be breached maliciously by HBAs where the intent is to disrupt a system.

25 Summary of the Invention

The present invention solves the problems discussed which are inherent with soft zoning systems by providing hardware-enforced zoning, also referred to herein as hard zoning. Hard zoning prevents breaching of assigned zoning by the accidental or intentional soft zoning problems discussed above, thereby
30 improving system data integrity and security.

Hard zoning solves the soft zoning problems by using a hardware check of the frame's Source ID (S_ID) both at the point the frame enters the fabric, and at the point the frame leaves the fabric.

As shown in Fig. 1, according to the invention, the frame S_ID is
5 validated at the point the frame enters the fabric (at an F_Port or FL_Port, but not an E_Port) to prevent incorrect S_IDs from breaching the table-driven zoning at the point where frames exit the fabric. If the S_ID is valid, it will enter the fabric and route normally. If invalid, the frame will not be routed but will be disposed of by the fabric according to FC rules.

10 The frame S_ID is validated at the point the frame exits the fabric (at an F_Port or FL_Port, but not an E_Port), to insure that the frame, based on the S_ID, is part of the zone or zones that the attached N_Port or NL_Port belongs to. Frame S_IDs are compared against a list of S_IDs that are valid for that port, where the size of the list is implementation-dependent, and where each entry
15 defines a source that is allowed to transmit frames to this destination. If the S_ID matches an entry in the list, it is routed out of the fabric to the destination. If the S_ID does not match any entry in the list, the frame is not passed to the destination, but is disposed of by the fabric according to FC rules.

This method works both for fabric clouds containing homogenous switch
20 devices connected together by E_Ports, and clouds containing heterogeneous switch devices connected together by E_Ports, providing all switch devices use S_ID-based hard zoning in the manner described above.

The method also partially works when some heterogeneous devices are equipped with S_ID hard zoning and others are not. In this case, the devices so
25 equipped are offered full hard zoning protection except from frames entering the fabric from devices not so equipped, whose S_ID is incorrect.

The method also works when in devices so equipped, some fabric ports are deliberately excluded from hard zoning protection, producing a deliberate half-duplex hard zoning case. This can be done to work around the limitations
30 of the Hard Zoning S_ID Inclusion List, which has a finite range.

The hard zoning methods and switches of the present invention provide a number of advantageous features, which include the following:

- The methods and techniques of the present invention provides a form of hard zoning in a switch fabric that is performed by hardware verification of frame S_IDs against an inclusion list of allowable S_IDs, before the frame is allowed to exit the fabric.
- 5 • The invention allows multiple overlapping hard zones for any destination (N_Port or NL_Port) across the largest of FC fabrics. The S_ID inclusion list is based on 24-bit addresses, allowing zones to be comprised of any nodes within the entire Fibre Channel 16,777,216 address space.
- 10 • The invention allows multiple overlapping hard zones to be implemented at the finest FC addressing granularity, which is down to the loop device, both at the frame source and the frame destination, where each of the 126 possible nodes on a loop can express unique zone characteristics.
- The invention accomplishes hard zoning at a fabric destination port
15 (F_Port or FL_Port) by an Inclusion Table, unique to each fabric port, of legal S_ID values. The Inclusion Table is used by hardware to pass legal frames, and bar illegal frames. This mechanism guards against intentional and un-intentional zone boundary violations.
- The invention implements the Inclusion Table as a programmable
20 hardware table, implemented as a CAM, containing multiple entries, each entry containing a 24-bit S_ID value, along with various comparison controls. A single entry typically represents a single source, such as an N_Port attached to an F_Port, or an NL_Port attached to an FL_Port.
- The invention provides a way to increase the range of sources an
25 Inclusion Table can express, by implementing wild cards, on an entry-by-entry basis, which can disable the comparison of the Port value, or the Port and Area values, of the S_ID. A single entry can then express all Ports within an Area, or all Ports and Areas within a Domain.
- The invention allows the flexibility of ranges to increase, by
30 implementing exceptions to a range. A single entry defines the range, and one or more entries define exceptions to the range.

- The invention prevents zone breaching via false S_IDs, by implementing S_ID validation against the ports native ID as frames enter the fabric. Valid S_IDs route normally, invalid S_IDs are not routed.
- The invention allows an F_Port and its attached N_Port to belong to as many zones as the S_ID inclusion list allows.
- The invention allows multiple zones per loop, and overlapping zones on a loop. The number of zones supported on a loop is variable (n), is at least 2, and may be as high as requirements allow. Any loop device (based on Port or AL_PA) can belong to 1 to n zones.
- The invention's restrictions on the total number of zones on an F_Port, or the total number of zones on a loop, does not place restrictions on the total number of zones in a fabric.
- The invention works across fabrics comprising a single switch element (Domain), fabrics comprising multiple homogeneous switches connected together by E_Ports, and across fabrics comprising heterogeneous switches connected together by E_Ports.
- The invention allows certain frame types to be unaffected by zoning. Frames with FC Well-Known S_IDs, and any frame sourced by the fabric itself, are excluded from zoning, i.e., they will always route.
- The invention allows known "ill-mannered" S_IDs to be recognized in hardware, allowing a different disposition for these frames, such as immediate tossing rather than being processed as an exception by the fabric manager.
- The invention applies both to duplex zoning, where bidirectional traffic hard zoning is enforced on both ends, and half-duplex zoning where hard zoning is enforced on one end but not the other. Half-duplex zoning can solve limitations of the S_ID Inclusion Table for some topologies, and is a requirement when a fabric cloud contains multiple switch types, where some employ hard zoning and others do not.

30 Brief Description of the Drawing

Fig. 1 is a symbolic representation of a FC fabric cloud, showing S_ID validation points according to one aspect of the invention;

Fig. 2 is a block diagram of a portion of a fabric showing source port S_ID validation according to one aspect of the invention;

Fig. 3 is a block diagram of a portion of a fabric showing destination port S_ID validation according to one aspect of the invention;

5 Fig. 4 is a diagram of a SIL (Source S_ID Inclusion List) entry which may be used in the validation of Fig. 3;

Fig. 5 is a block diagram of a portion of a fabric showing S_ID compare according to one aspect of the invention;

Fig. 6 is a block diagram of a portion of a fabric showing SIL entry
10 compare according to one aspect of the invention;

Fig. 7 is a diagram showing the Source Zone Mask format;

Fig. 8 is a diagram showing the Destination Zone Mask format;

Fig. 9 is a block diagram of the Hard Zoning State according to one aspect of the invention;

15 Fig. 10 is a diagram of an example (Example A) of the use of the invention in connecting multiple hosts to multiple storage devices, with hard zoning;

Fig. 11 is a diagram of an example (Example B) of the use the technique of wild carding, in connecting multiple hosts to multiple storage devices with
20 hard zoning; and

Fig. 12 is a diagram of an example (Example C) of the use the technique of wild carding with exceptions, in connecting multiple hosts to multiple storage devices with hard zoning.

Detailed Description of the Preferred Embodiments

25 The overall operation of S_ID validation is discussed above with reference to Fig. 1. Detailed operation is described below with reference to the Figures and tables.

Source Fabric Port S_ID Validation

Fig. 1 shows the technique for source fabric port S_ID validation. This
30 ensures that incoming frame S_ID values are legal, in order that the Hard Zoning mechanism will be effective. This feature requires an enable/disable control, because it should be enabled only on F_Ports and FL_Ports, but not on E_Ports.

The Native ID register (1) is resident to the fabric port logic, and is typically written by the fabric manager (15) to whatever ID has been chosen for the port, but may be hard-wired into the logic in a simpler design with greater operating restrictions. The Native ID is the Domain and Area of the F_Port or FL_Port. The Enable Hard Zoning FF (2) is also resident to the fabric port logic and written by the fabric manager (15), which enables the feature on F_Ports and FL_Ports, but disables it on E_Ports.

The incoming frame (3) is parsed and the S_ID contents (bits 23:0) of frame word 2 are presented to this circuitry. The frame S_ID Domain and Area (5), bits 23-8 of frame word 2, are compared against the Native ID Register (1) Domain and Area (6) in comparator (11). Match/mismatch is fed to gate (14).

If there is a miscompare, and Enable Hard Zoning (2) is active, then gate (10) specifies an invalid frame S_ID. In this case, the frame is not routed normally (it will never reach the intended destination), but is typically forwarded to the fabric manager for processing according to Fibre Channel rules for switches. This typically means that a Class 2 frame that fails the S_ID test spawns a response back to the sender with reason code, and that a Class 3 frame that fails the S_ID test is tossed.

If there is no miscompare in the frame S_ID, or if Enable Hard Zoning (2) is inactive, then gate (10) specifies that the frame will route normally, which means it will route to the fabric destination port specified by the frame D_ID.

This feature ensures that frames entering the fabric are not forwarded if the S_ID is illegal in any way.

Destination Fabric Port S_ID Validation

Destination fabric port S_ID validation, described with reference to Fig. 3, ensures that outgoing frames are legal to send to the attached device(s). This feature requires an enable/disable control, because it should be enabled only on F_Ports and FL_Ports, but not on E_Ports.

Issues with linear tables.

Ideally, S_ID hard zoning would be performed in each host bus adapter (HBA) attached to the fabric. Each HBA would have its own unique zoning

table. However, FC standards demand that zoning functions, hard and soft, be contained in the fabric.

A linear inclusion table would seem to be the most straightforward. However, to work for any address in the FC addressing range requires a very large table. For an F_Port, it would have an entry for every possible FC source address, which results in a depth of 256 Domains * 256 Areas * 256 Ports = 16,777,216 table entries/port. If an FL_Port, the table has to also represent up to 126 loop devices, each of which may belong to different zones, which would require a unique linear inclusion table for each. The total number of inclusion table entries for an FL_Port would then be $256 * 256 * 256 * 126 = 2,113,929,216$ table entries/port.

Shorter linear zone tables can be used, where a limited set of addresses, typically starting at the value Domain 1/Area 0, Port 0, and increasing sequentially up to the maximum table size implemented, can be economical. However, these tables cannot express any FC address and so greatly limits the fabrics that can be handled. For example, if the table had 512 entries, it could express all addresses in Domains 0x01 and 0x02, but none in Domain 0x03 through 0xEF.

The method described here utilizes a random table of 24-bit S_IDs, designed as CAM (Content Addressable Memory) where a frame S_ID can be compared against all CAM entries simultaneously seeking a comparison, or inclusion. Since the entries are 24-bit, the CAM can represent any FC address, but is limited to representing a subset of FC addresses far less than the FC maximum. For example, it could be expected that the CAM could economically hold 16-256 addresses, but could be less or more depending on the application and availability of resources.

The method has a single CAM per fabric port transmitter, even when the destination is an FL_Port, and so must work with as many as 126 loop destinations. For this reason, the CAM provides a source zone mask for each S_ID entry, which is compared against a destination zone mask contained in a separate lookup table.

There is a single destination zone mask table per fabric port transmitter, each entry representing a loop destination, or AL_PD, and which typically has an entry for every one of the 126 legal AL_PDs.

CAM-Based Inclusion Table.

- 5 This uses the S_ID Inclusion List (SIL), which describes which sources are allowed to send to this port, and if the destination is a loop, what zones each source belongs to.

It also uses the AL_PD Zone List (AZL), used only if the destination is a loop, which describes what zones a destination AL_PA belongs to.

10 The S_ID Inclusion List (SIL).

- SIL (31) is a programmable list of 24-bit S_IDs deemed by the fabric manager to belong to the same zones as the port in question, where each entry also has a 2-bit Compare Mask and a Source Zone Mask. SIL is written by the fabric manager (32) based on zone information. SIL (31) can be any size, but
15 typically would contain as many entries as is economically feasible, to allow as many sources as possible, and to cover as many topologies as possible. SIL (31) provides S_ID compare information (33) to S_ID Compare (34) and the source zone mask (41) to the Source Zone Mask Mux (42).

The Sil (31) entry format is shown in Fig. 4.

- 20 “Source S_ID” is 24 bits, and defines a legal source S_ID, if the “Compare Mask” value is not 00. Any S_ID within the FC address range of 0x0 - FFFFFFFF can be expressed.

“Source Zone Mask” is a bit mask of variable size, which defines which zone or zones the source belongs to. See Zone Mask explanation below.

- 25 “Compare Mask” defines how the compare against the frame S_ID is to take place

00 = Slot not valid. No compare is possible against this entry.

01 = No mask. Domain/Area/Port are compared [see note 1 below]

10 = Mask Port. Domain/Area are compared [see note 2 below]

- 30 11 = Mask Area and Port. Domain is compared [see note 3 below]

[note 1] The entry represents exactly 1 source.

[note 2] The entry could represent 126 sources.

[note 3] The entry could represent $256 * 126 = 32,256$ sources.

S_ID Compare.

S_ID Compare (34) (see Fig. 5) is used to compare a frames' S_ID simultaneously to all entries in SIL (31) and present the results to the Hard
 5 Zoning State Machine (47). Each SIL (31) entry is compared against the frames S_ID (23:0) (37), using the S_ID value and Compare Mask of the SIL (31) entry (33). When a match is made, S_ID Compare (34) encodes the selected entry number into a value (43) that controls the Source Zone Mask Mux (42). The Hit (45) and Multiple Hit (46) status is shipped to the hard Zoning State Machine
 10 (47) for processing.

S_ID compare (34) distills the Hit/not status from every SIL entry compare as described above into a No Hit, Hit, or Multiple Hit status for the frame being processed. It also encodes the entry number of the hit into a binary value for use in the Source Zone Mask Mux.

15 There exists an autonomous compare circuit for every SIL entry, as shown in Fig. 6. The Hit/Not Hit output feeds the Distiller (see Fig. 5).

Distiller

The Distiller processes the results of SIL entry compares.

A Miss occurs when neither Hit nor Multiple Hit occurs. This causes the
 20 frame to be rejected.

A Hit occurs if a single entry compares. This is the first step in causing the frame to be accepted, the second being the SIL/AZL zone mask compare.

A Multiple Hit occurs if two or more entries compare. This causes the frame to be rejected. This is the way exceptions to ranges is implemented, and it
 25 also covers programming errors.

Entry Number is used on a Hit, to be a mux select for Source Zone Mask Mux.

Source Zone Mask Mux

The Source Zone Mask Mux (42) produces the zone mask associated
 30 with the source S_ID that matched the frame S_ID. The mux select (43) is produced by S_ID Compare (34) which is the encoded value of the SIL (31) entry that matched the frame S_ID. The Source Zone Mask Mux output (44) is

given to the Hard Zoning State Machine (47) which will compare it against the destination zone mask (40).

The source zone mask can contain any number of bits, depending on how many zones a loop is designed to handle. For example, a 4-bit zone mask

5 implies that a loop can have up to 4 zones, and an 8-bit mask allows 8 zones.

The size of the source zone mask in SIL (31) must be identical to the size of the destination zone mask in AZL (38).

The AL_PD Zone List (AZL)

AZL (38) (Fig. 3) is a 126-entry programmable table, one entry for every
10 legal AL_PA, that contains the zone mask for each destination loop port. The frame AL_PD (36) provides the address to AZL (38), which produces the destination zone mask (40), which is given to the Hard Zoning State Machine (47) which will compare it against the source zone mask (44). AZL is written by the fabric manager (39) based on zone information. In order that the loop not be
15 restricted in the choice of assignable AL_PAs, it is desirable for AZL to have 126 entries, one for each legal AL_PA. Fewer entries are possible if restricting the AL_PA range is acceptable.

The destination zone mask can contain any number of bits, depending on how many zones a loop is designed to handle. For example, a 4-bit zone mask
20 implies that a loop can have up to 4 zones, and an 8-bit mask allows 8 zones. the size of the destination zone mask in AZL (38) must be identical to the size of the source zone mask in SIL (31).

Hard Zoning Enable

The Hard Zoning Enable (53)(Fig. 3) is a storage element programmed to
25 enable or disable hard zoning on the port, and is written by the fabric manager (54) based on zoning and topology information. If hard zoning is employed in the fabric cloud, this will be set active on F_Ports and FL_Ports, and inactive on E_Ports.

The state of Hard Zoning Enable (52) is fed to the Hard Zoning State
30 Machine (47) to condition the function. Hard Zoning Enable (52) is unique to the port, and is the same storage element that enables/disables S_ID validation on incoming frames.

The Hard Zoning State Machine

The Hard Zoning State Machine (47) (Fig. 9) determines the disposition of all frames attempting to exit the fabric at a particular port.

- If Hard Zoning is disabled;
 - 5 - All frames pass this function and, in lieu of other fabric functions, are forwarded out of the fabric to the destination node.
 - If Hard Zoning is enabled;
 - If the frame S_ID is in the range of 0xFxxxxx - FFFFFFFF, or if the frame originated in the switch box processor complex, hard
 - 10 zoning is ignored and the frame is routed normally.
 - If the port is an F_Port, only the S_ID compare is required. In this case, zone masks are logically irrelevant, but all zone masks in the SIL and all zone masks in AZL are set to some value (hex FF for example) that guarantees a zone mask compare in all cases.
 - 15 Note that the design could be implemented where an F_Port could disable the zone mask compare to avoid having to program the zone masks.
 - If the port is an FL_Port, the compare is the same, but the zone masks are now relevant and must be programmed according to the
 - 20 actual zones in use.
 - If there is a hit in SIL, but the SIL zone mask = 00, the frame is invalid. This is a deliberate way to reject known bad sources.
 - If there is a multiple hit in SIL, the frame is invalid. This implements the Exception To A Range function.
- 25 The Source Zone Mask (62) is provided by the Source Zone Mask Mux (42). The Destination Zone Mask (63) is provided by AZL (38). A bit-wise compare is made between the two masks in AND gates (66), all of which are ORed together in gate (67), which is active if any bit pair is set, and inactive if no bit pairs are set.
- 30 Function (68) is active if all source zone mask bits are zero. This is the test for a known bad S_ID that has been tagged in this way to insure its immediate tossing.

S_ID Hit (60) and S_ID Multiple Hit (61) are from S_ID Compare (34). Hard Zone Enable (65) is from the storage element of the same name (52). The translation for S_ID = 0xFxxxxx + Frame Source = Fabric Manager (64) is made locally.

5 AND gate (69) commands the unconditional toss of frames whose source
zone mask = 0.

AND gate (70) commands that a frame is valid because it passes the general hard zoning test.

AND gate (71) commands the general frame valid.

10 Example A:

Multiple Hosts On F Ports Connected to Multiple Storage Devices on FL Ports

The diagram in Fig. 10 depicts two hosts attached to a switch that is the access to their storage. Hard zoning is set up so that Host A can only access disks in zone A, and Host B can only access disks in zone B. Host A and Host B can also access each other, and Host A and Host B share access to some disks. Zone masks have bit 0 assigned to zone A and bit 1 assigned to zone B.

The programmable values for each fabric port is as follows.

Port 01,00 (Domain 01, Area 00): Host A

Table 1: Example A SIL Contents

20	SIL Entry	Compare Mask (bin)	Domain Value (hex)	Area Value (hex)	Port Value (hex)	Source Zone Mask (hex)	Comment
	0	02	01	03	xx	FF	Host B
	1	01	01	01	04	01	Loop x, alpha = 04 (belongs in zone A)
	2	01	01	01	05	03	Loop x, alpha = 05 (belongs in zones A) & B)
25	3	01	01	02	03	01	Loop y, alpha = 03 (belongs in zone B)

Table 2: Example A AZL Contents

5	AZL Entry (hex)	Dest Zone Mask (hex)	Comment
	All	FF	

Port 01,01 (Domain 01, Area 01): Loop X

Table 3: Example A SIL Contents

10	SIL Entry	Compare Mask (bin)	Domain Value (hex)	Area Value (hex)	Port Value (hex)	Source Zone Mask (hex)	Comment
	0	02	01	00	xx	01	Host A (belongs to zone A)
	1	02	01	03	xx	02	Host B (belongs to zone B)

15

Table 4: Example A AZL Contents

20	AZL Entry (hex)	Dest Zone Mask (hex)	Comment
	04	01	belongs to zone A
	05	03	belongs to zone A and B
	06	02	belongs to zone B
	Others	00	

25

Port 01,02 (Domain 01, Area 02): Loop Y

Table 5: Example A SIL Contents

5	SIL Entry	Compare Mask (bin)	Domain Value (hex)	Area Value (hex)	Port Value (hex)	Source Zone Mask (hex)	Comment
	0	02	01	00	xx	01	Host A (belongs to zone A)
	1	02	01	03	xx	02	Host B (belongs to zone B)

10

Table 6: Example A AZL Contents

	AZL Entry (hex)	Dest Zone Mask (hex)	Comment
	01	02	belongs to zone B
	02	02	belongs to zone B
15	03	01	belongs to zone A
	Others	00	

Port 01,03 (Domain 01, Area = 03): Host B

Table 7: Example A SIL Contents

5	SIL Entry	Compare Mask (bin)	Domain Value (hex)	Area Value (hex)	Port Value (hex)	Source Zone Mask (hex)	Comment
	0	02	01	00	xx	FF	Host A
	1	01	01	01	06	02	loop x, alpa = 06 (belongs to zone B)
	2	01	01	01	05	03	loop x, alpa = 05 (belongs to zone A and B)
10	3	01	01	02	02	02	loop y, alpa = 02 (belongs to zone B)
	4	01	01	02	01	02	loop y, alpa = 01 (belongs to zone B)

Table 8: Example A AZL Contents

15	AZL Entry (hex)	Dest Zone Mask (hex)	Comment
	All	FF	

Configured as described above, the desired hard zoning for Example A,
 20 Fig. 10 is achieved.

Example B: Wild Carding

The example in Fig. 11 depicts how a single wild-carded SIL entry can represent all devices in a Domain. This shows Host A's view of the hard zoning, where some of the other ports in the same Domain are in zone A, and all of the
 25 ports in the other Domain are part of zone A.

Port 01,03 (Domain 01, Area = 00): Host A

Table 9: Example B SIL Contents

5	SIL Entry	Compare Mask (bin)	Domain Value (hex)	Area Value (hex)	Port Value (hex)	Source Zone Mask (hex)	Comment
	0	02	01	01	xx	FF	device on Domain 1, Area 1
	1	02	01	03	xx	FF	device on Domain 1, Area 3
	2	03	02	xx	xx	FF	all devices on Domain 2

10 Table 10: Example B AZL Contents

AZL Entry	Dest Zone Mask	Comment
All	FF	

Configured as described above, the desired hard zoning for Example B, Fig. 11 is achieved. Note that this example shows how an entire Domain can be represented by one entry. This same method can be used to represent an entire Domain/Area with one entry, which allows all Ports to be represented with one entry.

Example C: Wild Carding with Exceptions

The example in Fig. 12 depicts the same topology as for Example B but with an exception to a wild card. This shows Host A's view of the hard zoning, where some of the other ports in the Same Domain are in zone A, and most but not all of the ports in the other Domain are part of zone A.

Port 01, 03 (Domain 01, Area = 00): Host A

25

Table 11: Example C SIL Contents

SIL Entry	Compare Mask (bin)	Domain Value (hex)	Area Value (hex)	Port Value (hex)	Source Zone Mask (hex)	Comment
0	02	01	01	xx	FF	device on Domain 1, Area 1
1	02	01	03	xx	FF	device on Domain 1, Area 3
2	03	02	xx	xx	FF	all devices on Domain 2
3	02	02	02	xx	FF	exception to Domain 2 (exception)

Table 12: Example C AZL Contents

AZL Entry	Dest Zone Mask	Comment
All	FF	

Configured as described above, the desired hard zoning for Example C, Fig. 12 is achieved. Note that 4 entries represent 7 sources. Note also that this shows wild carding a Domain with a single entry, but uses additional entries for exceptions to that Domain. This can also be used to wild card a Domain/Area, which covers all Ports, but use additional entries for each Port exception to the Domain/Area.

It will be seen from the above description that the present invention provides improved FC switch techniques, by providing robust hardware-enforced zoning for data integrity and security. While specific embodiments and techniques have been described, it will be appreciated that the invention is not limited to those specific embodiments, and that many variations are possible within the scope of the invention.

WHAT IS CLAIMED IS:

1. A method of hard-zoning protection for loop-level addresses in Fibre Channel switching, comprising:
 - 5 receiving a frame at a port of a Fibre Channel fabric;
comparing the S_ID of the frame to the native ID of the port, and based on the comparison, routing a valid frame to its destination F_Port or FL_Port;
at the destination port, comparing the frame's S_ID to an inclusion list of sources permitted under the zoning to transmit to the destination port and for
10 loop-level addresses, comparing the zone of the destination and the zone of the source;
for valid frames, transmitting the frame through the destination port to the attached device or loop of devices.
- 15 2. A method according to claim 1 wherein the step of comparing the frame S_ID to an inclusion list includes simultaneous comparison of S_IDs using a content addressable memory associated with the destination port.
3. A method according to claim 1 wherein there are multiple zones
20 per loop attached to the destination port in the comparison of source and destination zones.
4. A method according to claim 1 wherein there are multiple overlapping zones per loop attached to the destination port in the comparison of source and
25 destination zones.
5. A method according to claim 1 wherein the inclusion list can express wild card designations to disable the comparison of Fibre Channel Port value or Port and Area value.
30
6. A method according to claim 1 wherein the inclusion list may have an entry defining a range of S_ID values, and additional entries

defining exceptions to the range, and wherein the comparison of permitted S_IDs is based on the range and exceptions.

7. A Fibre Channel switch, comprising:
 - 5 a port connectable as a source port to receive frames;
an S_ID validator associated with the source port and operable to compare the frame S_ID to the native ID or the source port when used as a F_Port or FL_Port, and operative to route valid frames through the switch fabric;
 - 10 a port receiving a frame routed through the fabric as a destination F_Port or FL_Port having a unique inclusion table of valid S_IDs and zones for devices and loop devices attached to the port according to defined zoning;
a destination port S_ID validator operably associated with the
15 destination port to compare S_ID and zones of frames routed to it through the fabric against the inclusion list and to transmit only frames with valid sources and zones to the devices and loop devices attached to the port.
- 20 8. A Fibre Channel switch according to claim 7 wherein the destination port S_ID validator includes a content addressable memory operative to simultaneously compare the frame S_ID to entries in inclusion table.
- 25 9. A Fibre Channel switch according to claim 7 wherein the defined zoning includes multiple zones per loop attached to the destination port.
10. A Fibre Channel switch according to claim 7 wherein the defined zoning includes multiple overlapping zones per loop attached to the
30 destination port.

11. A Fibre Channel switch according to claim 7 wherein the inclusion table can express wild card designations and wherein the destination port validator is operable in response thereto to disable the comparison of Fibre Channel Port value or Port and Area value.

5

12. A Fibre Channel switch according to claim 7 wherein the inclusion table can express a designation defining a range of S_ID values and additional entries defining exceptions to the range, and wherein the destination port validator is operable in response

10 thereto accept the range but not the exceptions.

FIGURE 1

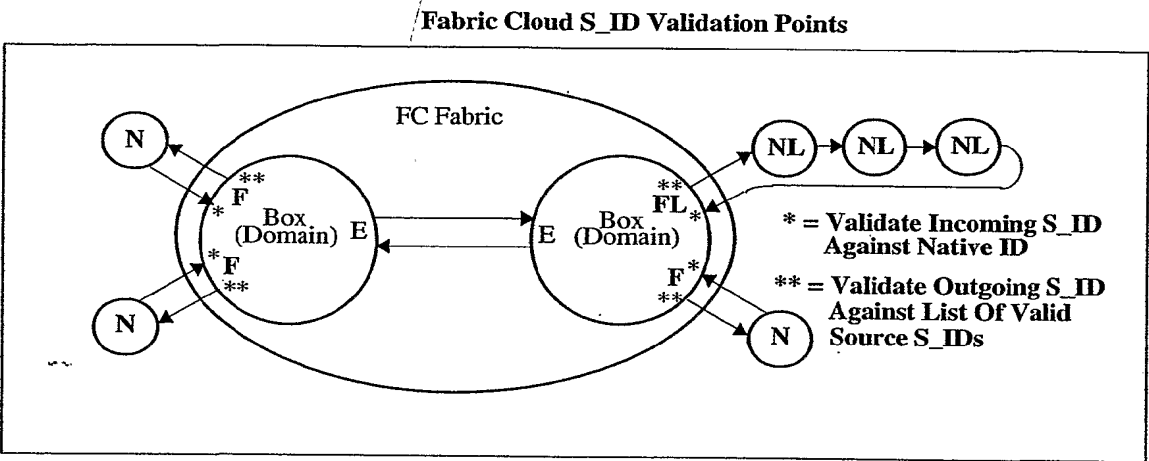


FIGURE 2

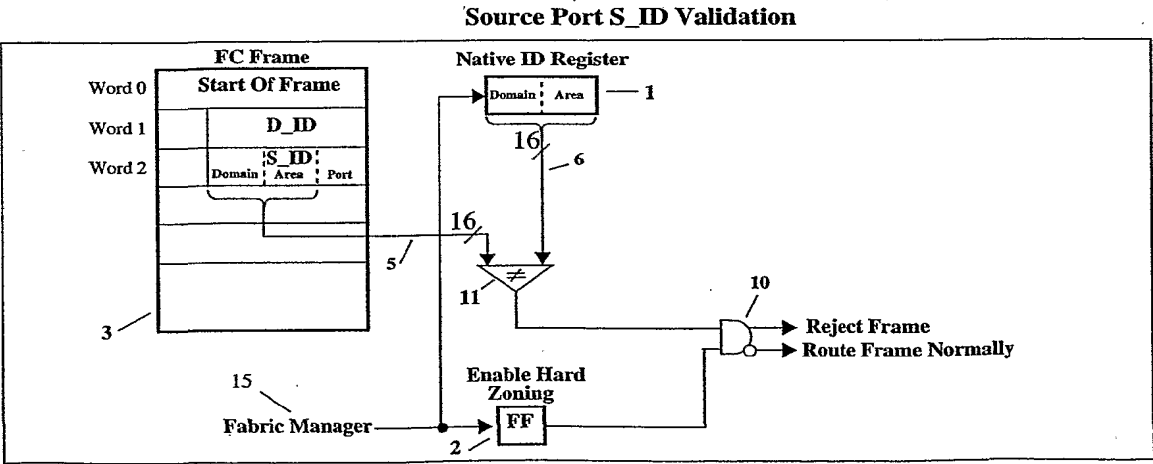


FIGURE 3

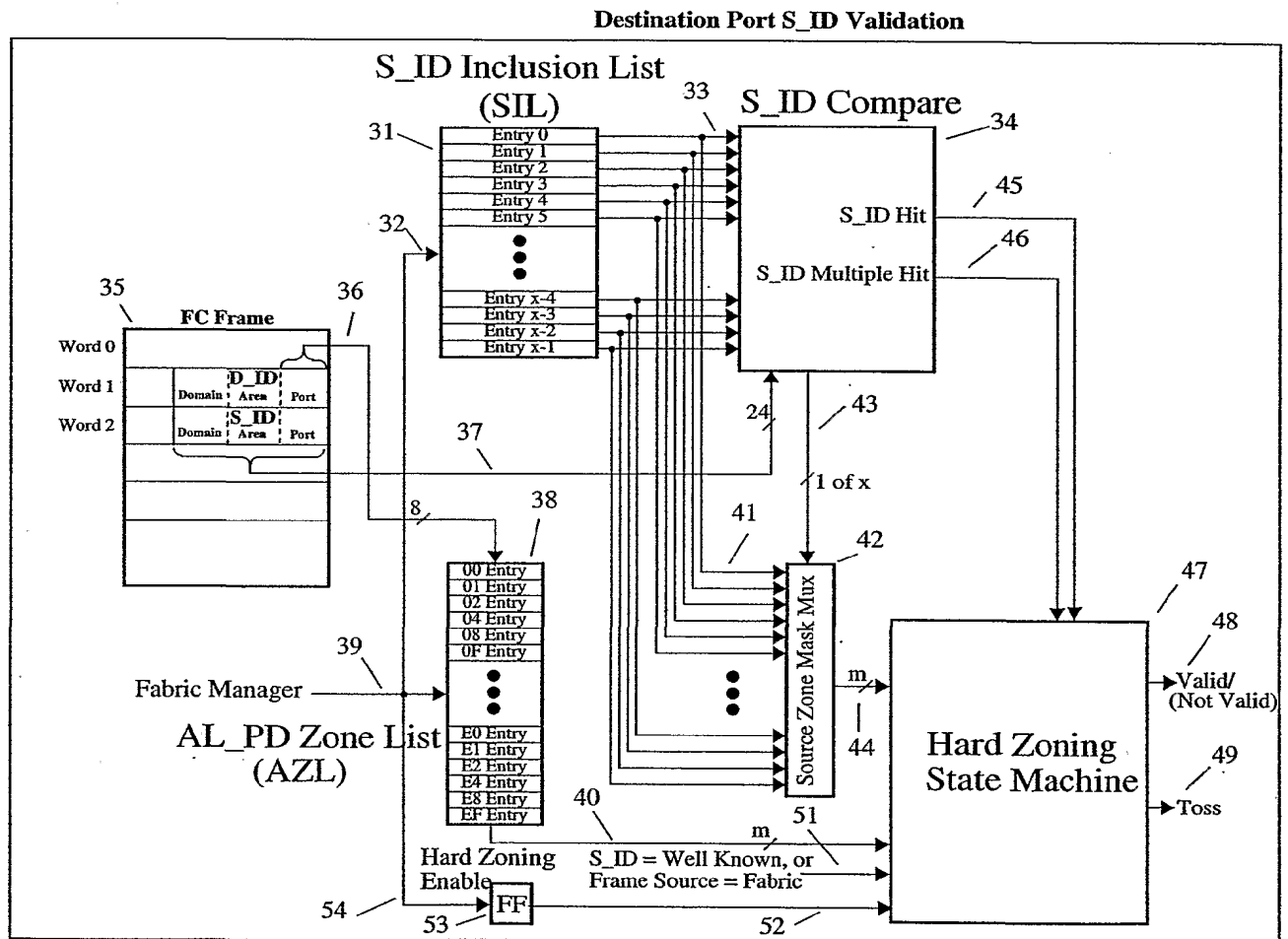


FIGURE 4

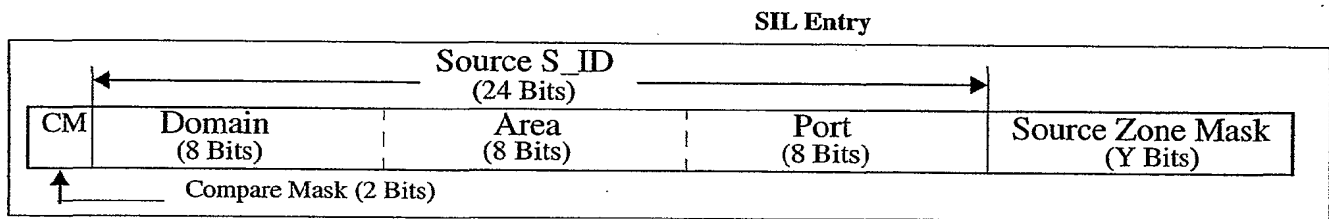


FIGURE 5

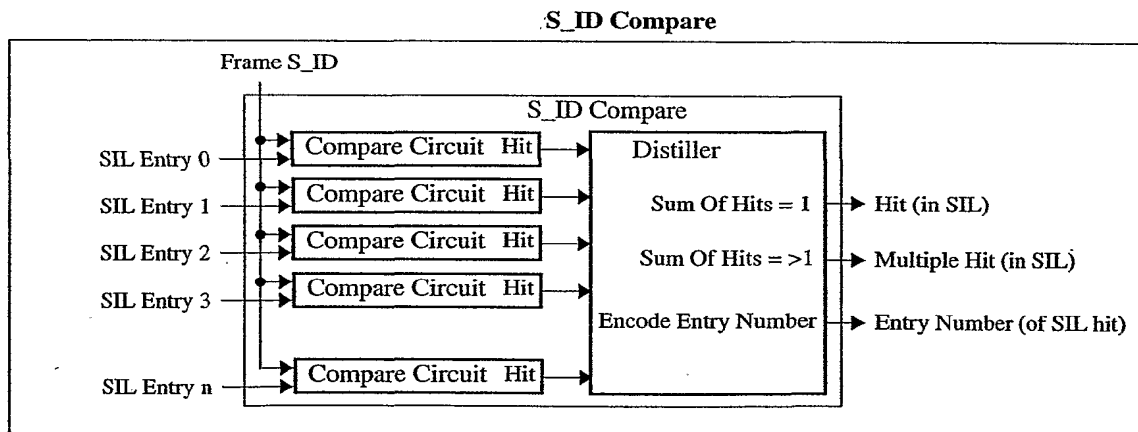


FIGURE 6

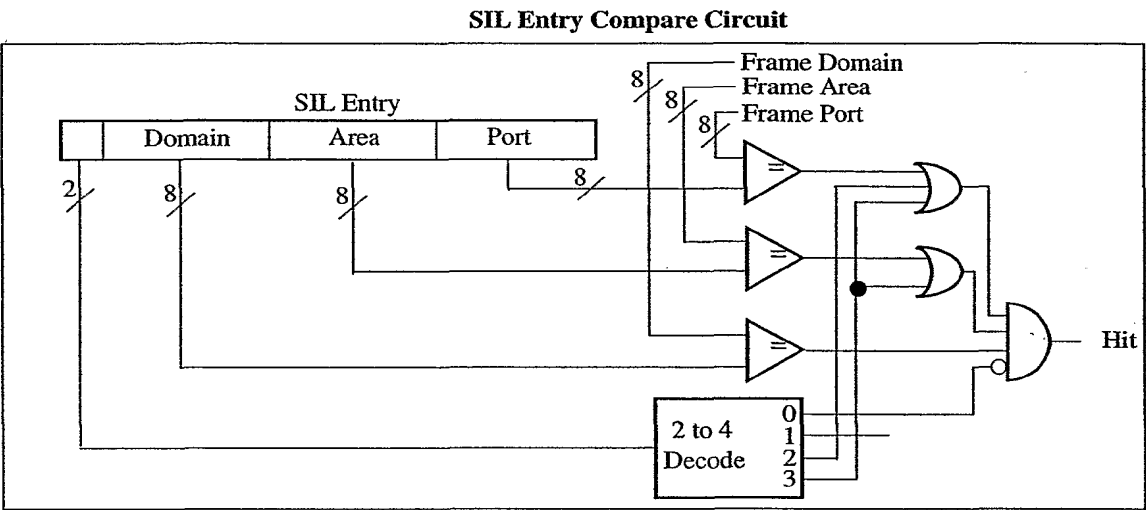


FIGURE 7

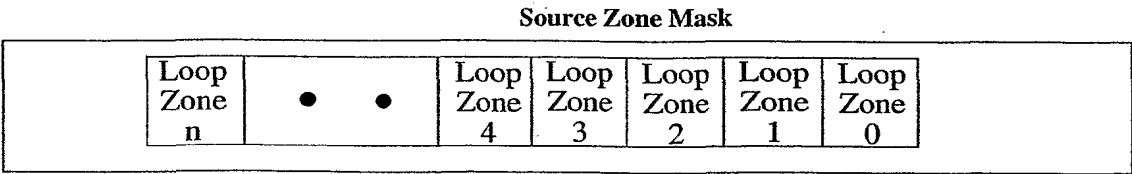


FIGURE 8

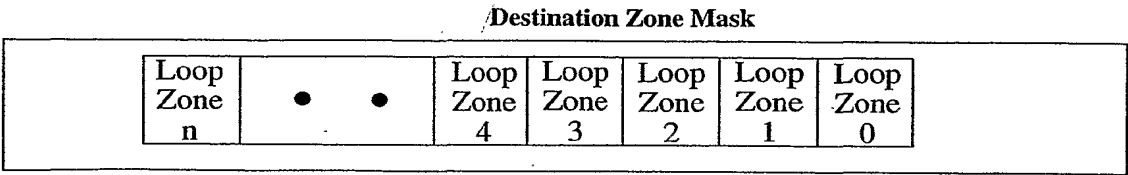


FIGURE 9

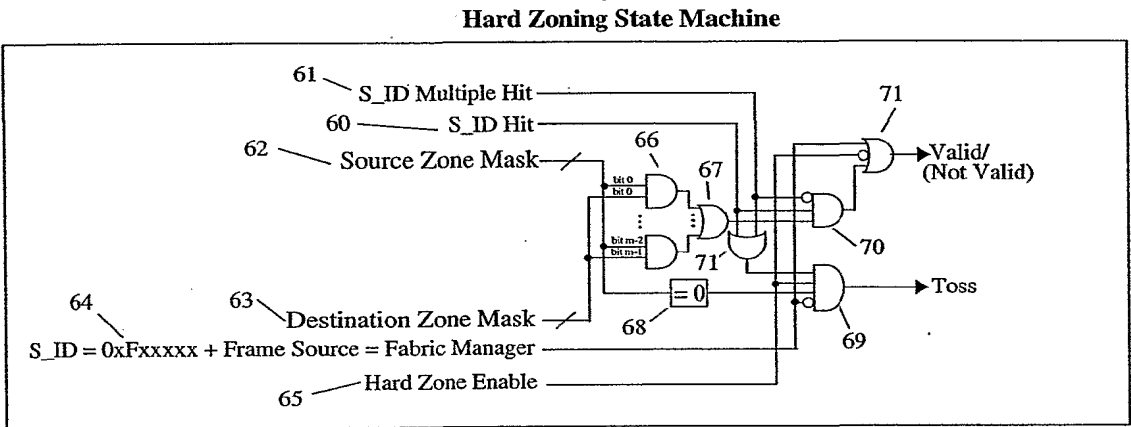


FIGURE 10

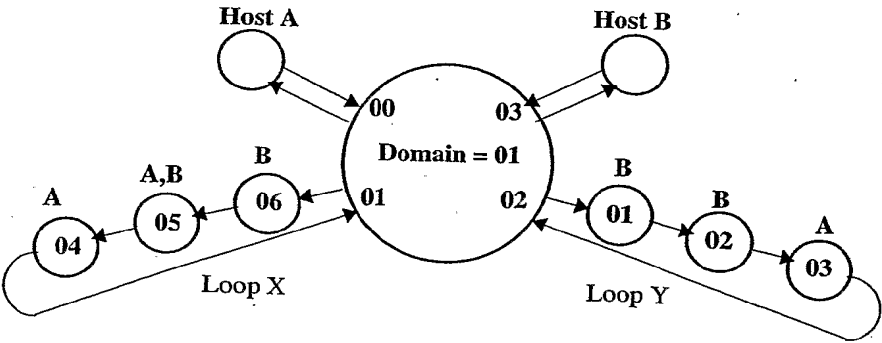


FIGURE 11

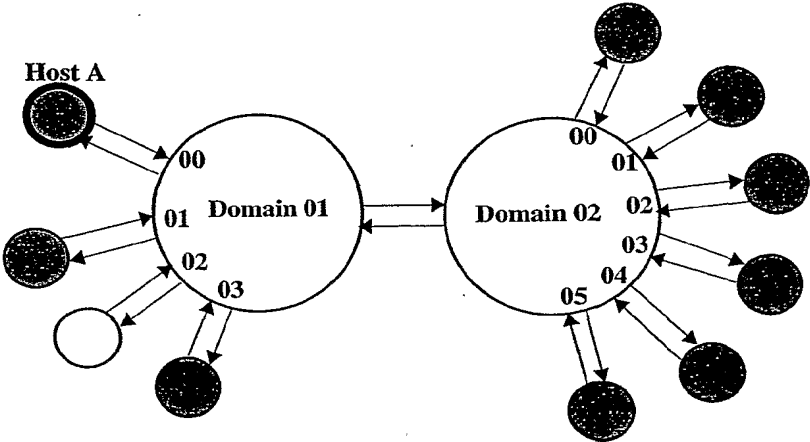


FIGURE 12

